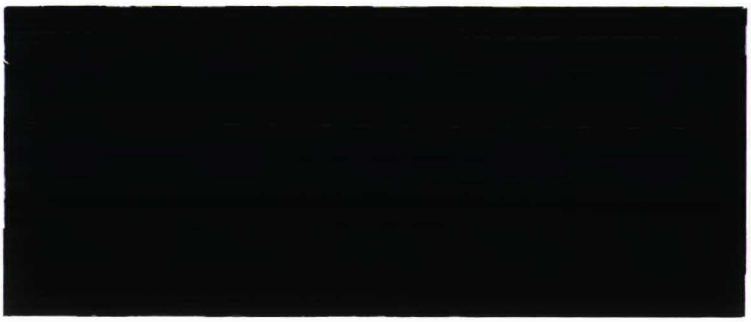
UNIVERSITEIT BRABANT
POSTBOX 90153
5000 LE TILBURG
THE NETHERLANDS

[redacted]

DEPARTMENT OF ECONOMICS

RESEARCH MEMORANDUM



INFOGAME USERS MANUAL

Rev 1.2 December 1988

Rommert J. Casimir

FEW 363

INFOGAME

USERS MANUAL

Rev 1.2 December 1988

Rommert J. Casimir

Rommert J. Casimir
Tilburg University
P.O Box 90153
5000LE Tilburg, Netherlands
R974CASI@HTIKUB5.BITNET

FOREWORD

Infogame (initially called Infolab) is a management game for research and education in the design of management information systems; it was to play a major role in a research project directed by Prof. Dr. Jack Kleijnen [1]. According to our original plans, details of the game would have been released only after extensive field testing with students and other user groups. For reasons which will be discussed elsewhere, however, it has been decided not to use Infogame in our future research. Because our efforts might have some use for other researchers, if only as a guide on how not to build a management game, we also decided to distribute the preliminary version of the manual and the game program.

1 R.J. Casimir: *DSS, Infomation systems, and management games.* Information and Management Vol 11 Nr 3 (Oct 1986), pp 123-129.

1 INTRODUCTION

Scope of this manual

This manual is intended for users of Infogame, a game for teaching the design and implementation of informations systems. It does not discuss the rationale for the game or any of its specific features. Infogame can be played as a regular game with up to twelve players or player teams, but it may also be used as an exercise by a single player. Section 6 describes the operation of the game; it is intended for game administrators and for players who play Infogame as an exercise. A full game administrator's manual will be published separately. Section 5 describes a standard set of data provided with the game; it must be replaced if the game administrator actually changes data. Because the game administrator can change a large number of parameters, the wording in the manual may be intentionally vague (for example: "there are up to five industries"). For educational and research purposes, a free copy of Infogame for the IBM-PC or compatible with a memory of 512K or more is available from the author; it can be obtained with the application form included with this report.

Type of game

Infogame contains elements from two types of games that are commonly used in management education: simulation games, where players take the role of middle managers making decisions in a limited area, such as production, and results are computed independently for each player or team, and managament games, where players assume the role of top managers unconcerned with the company's day to day problems, and results are primarily determined by competition with other players. In Infogame, operations are simulated in detail, and players get reports on those. However, players take on the role of top managers; they give instructions on operations in the form of rules, which may be interpreted as orders to middle management.

To clearly identify the difference between the three types of game we describe the way production is handled.

In *management games* , global production orders are given, such as "produce 10,000 tons of product X during this period" or "during this period, produce product X with 50 workers and 2 machines". The results are also in a global form, such as: "9500 tons of product X were produced during this period"; deviations from the instruction can be attributed to a global model, such as a LP model.

In *simulation games* that model a process industry, the player decides on production per day or per hour; results are reported on the same time scale. In simulation games that model a job shop, the player decides what job should be executed next whenever a state change (finishing a job, arrival of supplies, repair of a machine, etc.) allows execution of a new job; results of jobs are reported separately after the end of each job. In both types of simulation games, data are summed to determine the player's success.

In *Infogame* , production is controlled by a rule stating that a given amount (the order quantity) of product must be produced when the stock falls below a given level (the reorder level) and the requisite resources (materials, machines and manpower) are available; for each job, results are reported in detail. In an Infogame-like game that simulates a process industry, a rule would compute daily production from such parameters as size of stocks, number of operators available, and condition of installations.

Because both rules and other top management decisions (such as investment decisions) influence operations (for example, an incorrect relation between order level and order quantity will cause stockouts or excessive inventories), the player needs reports on actual operations. The contents of those reports are not defined in advance; the player must choose which events should be reported. For example, for most purposes it is not necessary to report both the allocation of materials at the start of a job and the use of materials at the end of a job.

The detailed reports, which may contain over 1000 records for a game round, would flood the player with detail. Consequently, an information system must be designed to process the raw data into usable information. This may be an accounting system that computes balance sheets and profit and loss statements, but it may also be a sophisticated decision support system that analyzes production and marketing data. We do not suggest a "best" approach. Some examples are given in section 4.5.

Apart from the data provided by the game program, the player receives data on the environment from the game administrator at the start of play (for example, attribute values of suppliers and machines). The only dynamic elements in the environment are the growth of the market and the labor force. No new markets, products, materials, machines or technologies are introduced during play. As a consequence, the game administrator will not supply new information on the environment once the play has started.

In contrast to most conventional management games, Infogame does not start with going concerns. Players start new companies from scratch; each new company starts with a cash capital of 10 million monetary units (say, dollars). In the first round, players must base decisions on the environment data; a prime reason not to start with going concerns was that we did not want to influence information system design by the form and contents of historical data.

Players must also decide on company structure: Infogame accomodates single-product companies as well as diversified and integrated firms. This was an additional reason to start without going concerns: the complexity of the production system influences the complexity of the information system, and the choice of the level of complexity should be left to the players.

How to use this manual

This manual is organized as follows: section 2 describes the system simulated by Infogame; it provides background information for making decisions in the game. Section 3 describes how decisions are actually entered; it may be necessary to turn to section 2 to find the implications of decisions or to section 5 to find the relevant data. Section 4 describes the output from Infogame. It should be consulted before decisions are entered if the player wants to make a decision on the data that should be reported; alternatively, she can ask for all data and subsequently study section 4 with the report at hand. As mentioned before, section 5 may be replaced by the game administrator (or by the player experimenting with Infogame), and section 6 can be skipped by players in a regular playing session.

2 OUTLINE OF THE GAME

2.1 Definitions

Each player or team of players manages a company producing goods or services in up to five competitive industries (services are characterized by the fact that no stocks of services can be held). There is no competition between products in different industries. Consequently, if the number of players is less than or equal to the number of industries, and each player specializes in a single industry, there is no competition. The game administrator may model industries after real-world industries. For example, the building industry produces to order only; the detergent industry has high advertising outlays; aerospace companies have a complicated production system, etc. In this manual, we only mention the abstract characteristics of industries and production processes: it is up to the player to identify an industry that can produce both for stock and to order, has high setup costs, and demand for a wide quality range, with the car industry.

A company employs a number of workers, it owns a number of machines of different types, it markets a number of products, and it buys, stocks and uses materials of different types. In each industry, one or more production technologies can be adopted. A technology specifies the quality of a product and the resources (machines, labor, and materials) needed to produce it. For example, in an industry there may be three technologies: a capital-intensive and a labor-intensive technology to produce a low-quality product, and a technology to produce a high-quality product. In earlier descriptions of Infogame, the word "process" was used instead of "technology"; it is still used in the program description.

Production is done in batches; so Infogame simulates job shops rather than process industries. Materials to produce a batch must be present at the start of a job and finished products become available only when the job has been finished. Consequently, a large batch size increases the size of the stock of materials and the stock of finished products. On the other hand, a small batch size may entail high setup costs. The player can also choose a technology that reduces the influence of batch size on results.

2.2 Time

In contrast to conventional management games, where all events during a gaming round are assumed to occur at the same moment (say, the start of the first day of the year), Infogame maintains an internal clock to time all events in a simulated period. However, it is not a real-time game, because the player cannot directly influence events after the start of a round. In the standard version, a playing round is equivalent to a quarter, containing 60 working days. Days are numbered continuously; for example, 362.14 indicates a moment in the third day of the third quarter of the second year. Most events can occur at any time of the day. Payments to suppliers and inventory taking occur only at the start of a day; salary payments, changes in the number of employees, and advertising outlays and interest payments occur only at the start of a month (i.e. at days 0, 20, 40 etc.).

In Infogame, long-term planning decisions (investment, employment, and marketing decisions) are made by the player; short-term scheduling decisions (ordering materials, starting a production run, delivery to consumers)are made by the game program. So players play the role of top managers, whereas the decisions of middle management and operating personnel are programmed. This "program" is defined in the simulation code, which executes such rules as "If a resource becomes available, the oldest production order waiting for that (and only that) resource will be executed". or "If the stock of materials (minus existing orders) falls below the player-specified reorder level, an amount equal to the player-specified order quantity is ordered".

The primary report from Infogame contains notices on states and events in the order of the time when they were made. For example, if a sale was made at 23.57, a machine was repaired at 23.61 and a new sale was made at 23.64, the events are reported in that order. The primary report reflects the outcome of strategic, tactical and operating decisions. It is up to the player to find the information necessary for his decisions in the next round.

2.3 Production modes

An industry in Infogame operates in one of the following modes:

a) Production for internal use only

The product is a material that cannot be sold on the market, but may be used in the production of other materials and/or consumer products. Production of materials is governed by the rules for acquisition of materials. For example, if the reorder level for material "MAT160" is set to 5000, and the order quantity is set to 2000, a production order for 2000 units of an internally produced material with quality index 160 is given when its stock is below 5000. When a job has been finished, the material produced is added to the stock of materials. A detailed description is given in section 3.10.

Whereas the quality index in other industries is a rough index of consumer acceptance, quality of a product used as a material should be exactly equal to the specified quality of the material. For example, if a material with quality index 120 is specified, a product with quality index 119 or 121 cannot be used. An example can be found in the car industry: a consumer may prefer a larger model for the same price, but parts for a specified model should conform to exact specifications.

b) Production for stock only

A production order is placed when the stock of finished products falls below the reorder level. When a production order is finished, the amount produced is added to the stock. Orders that cannot be filled from stock are disregarded; consumers just disappear or switch to competing products. For this mode, the reorder level should be positive; otherwise production will never start.

c) Production to order only

When a customer orders a product, a production order for the amount ordered is placed. In this context, "amount" is quivalent to "size": the demand for a building with a height of 320 feet cannot be filled by two buidings with heights of 300 and 20. When an order is met in time, it is delivered to the customer; otherwise all resources spent in producing it are wasted. This penalty may seem exaggerated, but it stresses the importance of planning in this type of industry. Moreover, when it is clear that an order will not be met in time before the start of production, no resources are spent on the order.

d) Production for stock or by order

Production is for stock, but customer orders that cannot be met from stock are retained. The amount produced is added to the stock, but existing back-orders that can be met in time are filled before the product is made available to new customers. Production orders are placed as long as the stock is below the reorder level. With this mode, a zero or negative reorder level is feasible. In that case, production is started only when there is a sufficient order backlog.

Production mode and other industry characteristics, such as average order size and relative importance of marketing instruments, are defined by the game administator.

2.4 Company design

Because all companies in Infogame start from scratch, players are free to decide which type of company they want to operate. Some examples are:

a) Single-product company

The success of a single-product company mainly depends on the choice of product and technology. Its operation entails two relatively simple tasks: Ensuring efficient production and selecting an optimal marketing mix. Information processing in this type of company is simple as there is no need to compute results for different products or technologies.

b Diversified company

A diversified company produces more than one product and/or applies more than one technology. When the relation between technologies and products is one-to-one, and each set of resources is applied to one product only, information processing is still relatively simple. However, if products are produced with several technologies, and/or share resources, intricate planning, scheduling, and accounting problems arise.

c Integrated company

An integrated company produces materials as well as final products. This poses problems in planning and in monitoring the overall efficiency of the production process, especially if the material is used in several products. Accountants will stress the importance of the correct valuation of stocks of intermediate products; there is, however, no common opinion on "correct valuation" in the accounting and auditing professions.

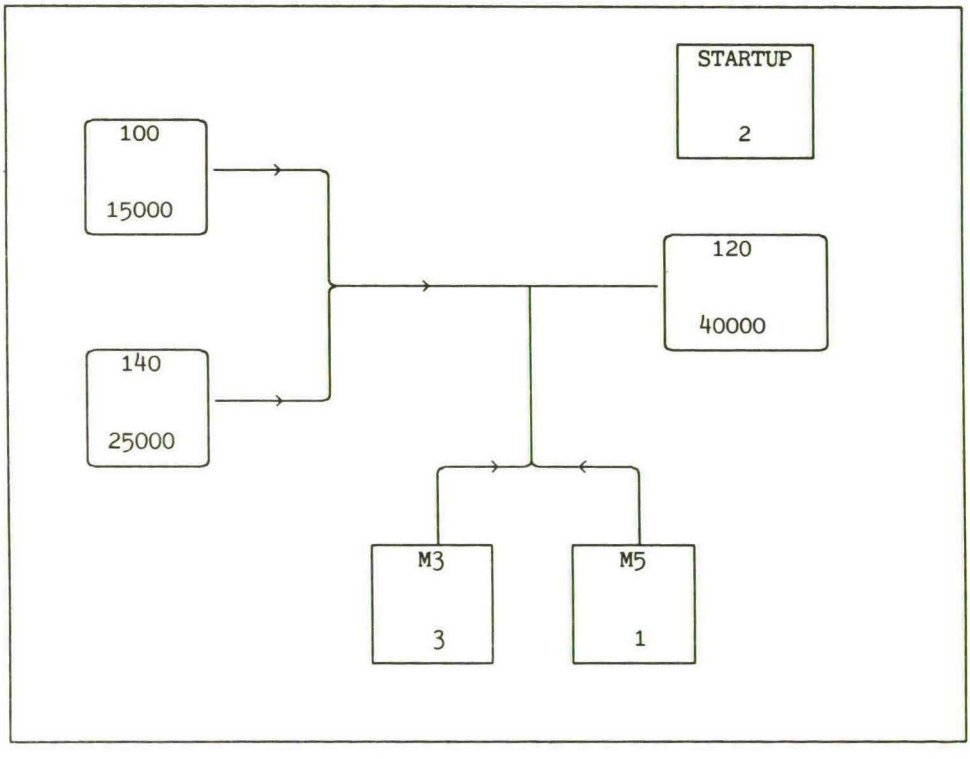
2.5 Production

For each product, one or more technologies must be chosen (see 3.7). Technologies that produce the same product must have the same quality index and must be in the same industry. For marketing purposes, one technology may be used to produce different products. Technology data are defined by the game administrator; actual values are given in section 5.

Variable	Description
Name	A unique name for the technology.
Industry	Name of the industry where the technology is used.
Quality	Quality of a product produced with this technology.
Operators	Number of fully productive operators needed for operation at full capacity.
Capacity	Maximum number of units that can be produced in a quarter at full capacity.
Startup	Time needed to start up production after a stop or a change in production setup.
Material	Type and amount of each material needed for production at full capacity during a quarter.
Machines	Type and number of each type of machines needed for production at full capacity.

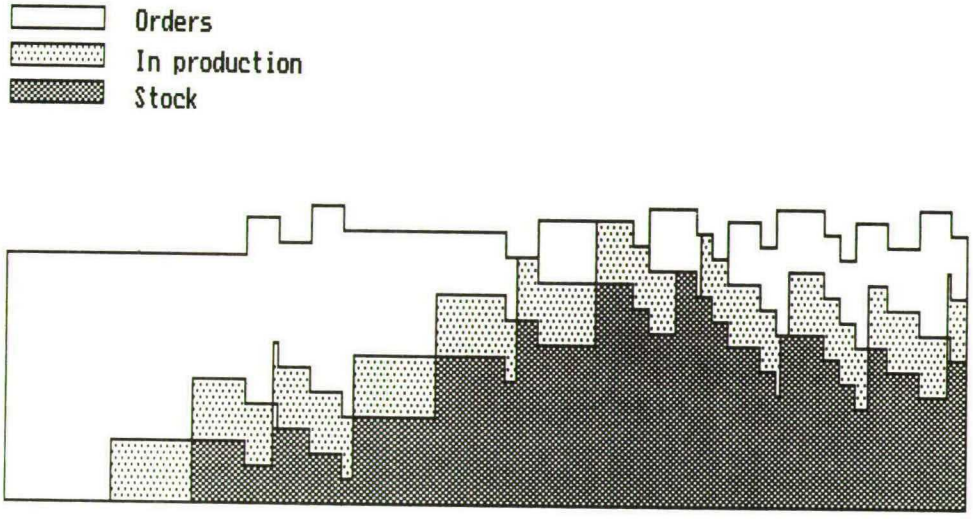
Example	
Variable	Value
Name	TRADB
Industry	TRAD
Quality	120
Operators	50
Capacity	40,000
Startup	2
Material	15000 of quality 100, 25000 of quality 140
Machines	3 M3, 1 M5

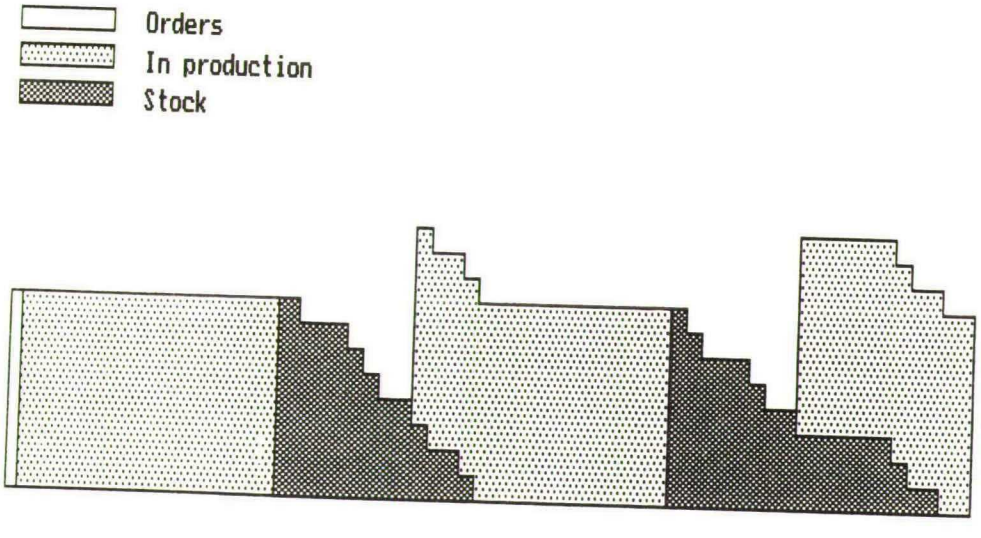
Maximum production and amount of materials needed (for maximum production) are flow data for a game period; the other data are state data. In the example, 40,000 units of a product in industry TRAD with quality index 120 can be produced if the company employs 50 fully productive operators, operates 3 machines of type M3 and 1 machine of type M5, and has sufficient materials of quality 100 and quality 140 in stock. This technology can be depicted as follows:



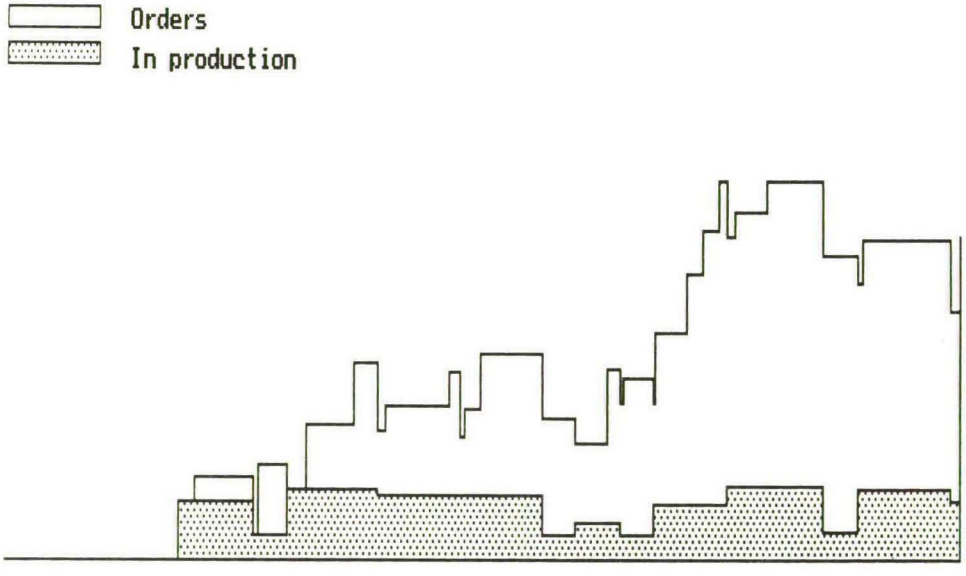
Production is done in batches. In industries with production for stock, scheduling is determined by two variables: *reorder level* and *order quantity* . Whenever the stock (including orders in process) is lower than the reorder level, an order to produce a batch with the size of the order quantity is entered into the order queue. An order in the queue is executed when it is the first for which all resources are available.

Consequently, production will be interrupted if the order level is lower than the order quantity. The graphs below show the size of stocks and orders over time. In the first graph, the reorder level is larger than the order quantity, in the second graph the converse is true.





For industries with production to order, an order from a consumer is entered in the queue of production orders when it is received; in this case, the variables "order quantity" and "reorder level" are not used. The first order from the queue for which all resources are available is executed. The graph below shows the production process.



The player also sets the maximum time allowed for production of a batch, a variable that is well known in computer operating systems (maximum job time). In industries with production to order, an order is not executed if production time would exceed the specified maximum time; in industries with production for stock, the batch size is reduced to what can be produced in the specified time.

Production is started only if enough materials are available; partial production of a batch because of lack of materials is not possible.

Mathematically, this is expressed by the condition:

$$\forall\, i \in M:\ q_i \geq \frac{r_i\ B}{C}$$

q_i Stock of material of type i.

r_i Materials of type i needed for production at full capacity.

M Set of types of materials.

B Batch size.

C Production capacity for a quarter.

Example					
r_1	r_2	B	C	min q_1	min q_2
20000	10000	1000	40000	500	250

If sufficient materials are available, net production time in days (without startup time) for a batch is the lower of the maximum time allowed and batch size divided by the actual capacity. Actual capacity is the product of full capacity and the minimal proportion of actually available to specified resources; it is never higher than full capacity. This is expressed by the formula:

$$t = \min\left(\frac{B.n}{\min\left(\frac{w}{W}, \frac{m_i}{M_i}\ i \in I, 1\right).C}, x\right)$$

t Net production time for a batch in days.

B Batch size.

w Number of fully productive operators.

W Number of operators needed for production at full capacity.

m_i Number of machines of type i in operation.

M_i Number of machines of type i needed for full capacity.

I Set of machine types.

C Production per quarter at full capacity.

n Number of days per quarter.

x Maximum time specified.

Examples										
B	w	W	m_1	M_1	m_2	M_2	C	n	x	t
1000	100	150	2	4	2	2	40000	60	5	3
1000	100	50	5	4	3	2	30000	60	5	2
3000*	100	100	5	4	3	3	30000	60	5	5

* Specified batch size. Actual batch size with production for stock is 2500; production will not be started with production to order.

The variables "batch size" and "maximum time" can be combined in three ways. First, maximum time can be given a value that is so high (say 60) that it never limits production time. Second, for an industry with production for stock, it can be set so low that production time, and consequently batch size, is always determined by the maximum time allowed. In this case, with the symbols from the formula to compute net production time, batch size is computed by the formula:

$$B = \frac{x.\min(\frac{w}{W}, \frac{m_i}{M_i} \quad i \in I \ , \ 1).C}{n}$$

Third, batch size can be used to determine regular production time, and maximum time is used as a "safety valve", just as maximum job time is specified by a computer center user to prevent the waste of valuable resources because of a program bug.

If a production job immediately follows another job producing the same product with the same machines, no extra startup time is needed. In all other cases, for example if production is stopped for any time, or if one of the machines is found defective, startup time is added to net production time. Consequently, unreliable machines not only directly decrease the capacity, but they also increase startup time.

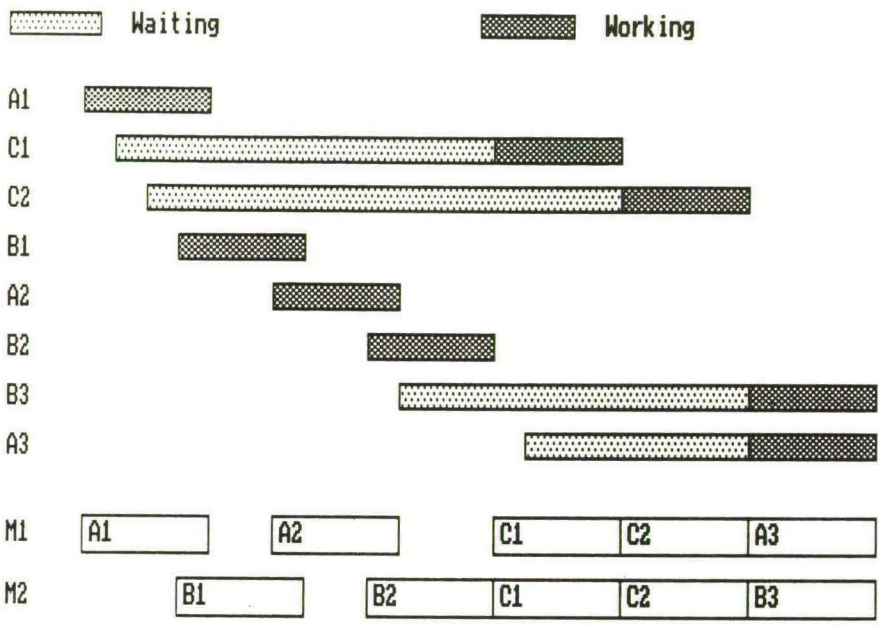
Total production time is computed by the formula:

T = t	If production is continued immediately with the same machines.
T = t + s	Otherwise

T Total production time
t Net production time
s Startup time

The mechanism that selects orders for production is derived from operating systems practice. When the stock is below reorder level (in industries with production for stock) or a consumer order is received (in industries with production to order), a production order is entered in a queue. When an order is added to the queue or a resource has become available (for example, materials or machines have arrived, machines have been repaired or a production job has been finished), the order queue is searched sequentially for the first production order that can be executed with the resources. Resources are allocated to this job, and the queue is searched for orders that can be executed with the remaining resources. Consequently, if many machines are present, a large number of jobs can be executed in parallel. As is known from operating systems theory, deadlock is impossible with this discipline, but a process may be "starved" by a number of other processes, each of which uses a subset of its resources. This may be clarified by the following example:

There are three processes: A (with orders A1,A2,A3), B (with orders B1,B2,B3) and C (with orders C1,C2). The only critical resources are machines M1 and M2. A uses M1, B uses M2, and C uses both M1 and M2. The graph on the next page shows the moment when the orders are entered in the queue, the time the orders where in the queue, and the actual execution of the corresponding jobs. If there had been a steady stream of orders from processes A and B, execution of C1 and C2 would have been postponed indefinitely. The discussion whether such problems should be solved by changing the scheduling algorithm or by better planning of resources is outside the scope of this manual.



All workers in a company have the same productivity index, a number between 0 and 1, which denotes the productivity of a worker as a fraction of the productivity of a standard (fully productive) worker. When a company employs its first workers, productivity is equal to the productivity of new workers. In subsequent periods, productivity of workers already with the company increases with the product of the learning factor and the productivity gap (the difference between 1 and the productivity). Total productivity is equal to the weighted average of the productivity of workers already with the company and newly employed workers.

Mathematically, productivity is defined by:

$$p_i = \frac{Wq + N(p_{i-1} + \ell(1-p_{i-1}))}{W + N}$$

p_i Productivity index for period i

q Productivity of new workers

W Number of new workers

N Number of workers already employed

ℓ Learning factor

Example					
i	N	W	q	ℓ	p
1	0	50	0.5	0.3	0.5
2	40	10	0.5	0.3	0.62
3	45	5	0.5	0.3	0.7106

3 HOW TO PLAY

3.1 Input formats

Infogame is normally operated in an interactive way: players enter decisions from the keyboard; they can correct inputs until the game administrator signals the end of the round.

There are three types of input:

a) Selection: Selection of an option from a list.
b) Set selection: Selection of a set of options from a list.
c) Table filling: Filling a table on the screen.

During selection and set selection the screen contains a list of options. During table filling a table appears on the screen; columns are headed by explanatory captions. In all cases, a summary explanation of the instruction is given in a footline. The following keys may be used:

Character	Selection	Set Selection	Table filling
→ Right arrow	Next item	Next item	Position right
← Left arrow	Former item	Former item	Position left
↑ Up arrow			Row up
↓ Down arrow			Row down
← Backspace		Remove selection	Remove left character
→| Tab			Column right
|← Back Tab			Column left
Ins			Insert space
Del			Delete character
↵ Enter	Select	Select item	Goto start of next row
F1	Help	Help	Help
F9	End selection	End selection	End filling
ESC	End selection	End selection	End filling

During table filling, fields that cannot be changed are protected. A protected field cannot be reached by the cursor.

3.2 Starting the program

The input program for Infogame is started from MS-DOS with the instruction:

INFOGAME

First a company is selected with a selection instruction:

```
ABC ========

CHOOSE OR CREATE COMPANY
```

An existing company is selected from the list, a new company is added by replacing the string "========" with a company name of eight letters or less. After choosing a company, a password must be entered by filling a table. For a new company, the same password must be entered both in the in the ENTER and in the VERIFY column:

ENTER	VERIFY

```
VERIFY PASSWORD
```

For an existing company, the password must be entered in the ENTER column:

ENTER

All characters are echoed with "X" on the screen. The program stops after three unsuccessful attempts to type the correct password.

```
ENTER PASSWORD
```

When the correct password has been entered, one of the following options is selected:

```
REGISTER INVEST SCRAP DESIGN REMOVE PRODP MAT SUPPL EMP BANK REPAY CREDIT

CHOOSE INSTRUCTION
```

The input session is ended by pressing the END key (F9 or ESC). After the return to MS-DOS, INFOGAME can be started again to correct data or to enter data for another company. Results are computed when data are entered for all companies in the game.

3.3 Instructions

The following instructions can be selected:

Instruction	Meaning	Default	Section
REGISTER	Define output signals	U	3.4
INVEST	Invest in machines	N	3.5
SCRAP	Scrap machines	N	3.6
DESIGN	Design a product	N	3.7
REMOVE	Remove a product	N	3.8
PRODP	Define product characteristics	U	3.9
MAT	Set parameters for materials	U	3.10
SUPPL	Choose suppliers	U	3.11
EMP	Define terms for employees	*	3.12
BANK	Ask loans	N	3.13
REPAY	Repay loans	N	3.14
CREDIT	Accept credit and set credit terms	U	3.15

The default code determines what will be done when the instruction is not selected. There are three options:

U Unchanged; the entry from the former period is copied.
N No action.
* Mixed: No employees hired or fired, other variables left unchanged.

For example, if no **SUPPL** instruction is given, suppliers for materials are not changed; if no **DESIGN** instruction is given, no new products are designed.

3.4 Define output signals (REGISTER)

The **REGISTER** instruction allows the player to define the extent of the output file. To this end, the player first uses set selection to select the events that will be reported and the inventories that will be taken.

The cost of reporting each event or inventory is defined by the game administrator and supplied to all participants. A description of events and inventories, and the syntax of the resulting output file is give in section 4. The list of events and inventories appears on the screen as:

```
matorder mat nomat alloc use paymat order prodqueu prodord prodrdy
deliver nodeliv salepay salary advert removep install noinst machall
machfree scrap defect repair interest hire fire nohire leave arrive
reinst ask assignw freework persadv mats ps machw machnu pers ops

WHAT REPORTS
```

Some data are redundant; for example, if the use of materials is reported at the start of a job (alloc), there is no need to report it when the job has been finished (use).

If one or more inventories have been selected, a table instruction is used to define how often those inventories will be taken. In addition to inventories listed in this table, the daily cash balance and the total accounting costs are reported without cost. If the REGISTER instruction is not used, or no items have been selected with the instruction, the report only contains the cash balances.

The format of the instruction to define report frequency is:

REPORT	FREQ
mats	10
ps	10
machw	10
machnu	10
pers	10
ops	10

SET REPORT FREQUENCY

Report frequency is given in days. For example, "mats 10" means that the stock of materials is inventoried every 10 days. As there are no discrepancies between inventory data and event data because of frauds, waste or errors, the only reason to collect inventory data as well as event data is to check on the information system. For example, product stock (ps) can be computed from production and delivery data if those are recorded. On the other hand, if product stock and deliveries are recorded, production may be computed from these data.

3.5 Invest in machines (INVEST)

The screen format for this instruction is:

MACH TYPE	NUMBER	REMARKS
M1	5	
M2	7	
M3	2	TYPE UNKNOWN

INVEST IN NEW MACHINES

When a company wants to purchase new machines, it defines the machine type and the number of machines. A machine will be installed only if it can be paid at the moment of delivery. When it is installed, it is paid immediately, it is given a sequential number, which will henceforth be used to denote this machine, and it can start producing. However, it cannot be used in a current production job. For each machine type the following data are given. Actual delivery times, failure times and repair times are drawn from a negative exponential distribution.

Machine	Unique name of the machine type
MTTD	Mean time to deliver in days
Repair	Cost of repair per day
Price	Price of one unit
MTBF	Mean time between failures in days
MTTR	Mean time to repair in days
DELTA	Decrease in MTBF and increase in MTTR per period

When alternative technologies are available to produce a product, cost and reliability of the machines employed may be a decisive argument. Another reason to choose a particular set of technologies may be that they use a common set of machines. Such decisions have to be based on data supplied by the game administrator before playing. However, game reports should indicate whether plans have been realised.

3.6 Scrap machines (SCRAP)

In the **SCRAP** instruction, the player selects a machine to be scrapped from the machines purchased in preceding quarters. Accordingly, it cannot be used in the first quarter. A machine should be scrapped when its repair costs surpass its utility. Expected repair costs may be computed from the data specified in section 3.5; actual repair cost may be given in the report. If a machine is not in use, it is scrapped at the start of the quarter, otherwise it is scrapped when the job using it is finished. The machines to be scrapped are selected with a set selection instruction from a list, where machines are indicated by their numbers (see 3.5):

```
1 2 3

SCRAP MACHINES
```

3.7 Design a product (DESIGN)

The **DESIGN** instruction uses the following screen format:

PRODUCT	PROCESS	PROCESS	PROCESS	PROCESS	REMARKS
P1					NO PROCESS GIVEN
P2	SPECA	TRADA			DIFFERENT INDUSTRIES
P3	TECHA	TECHC			DIFFERENT QUALITY
P4	TRADA				

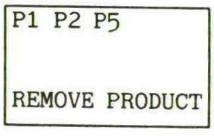
```
DESIGN NEW PRODUCT
```

The **DESIGN** instruction adds a new product and defines one or more technologies to produce it. The technologies in turn define the industry and the quality index of the product. If more than one technology is

specified, all should be in the same industry and have the same quality index. When applied to an existing product, the DESIGN instruction can be used to add or remove a technology. For example, a labor-intensive technology may be replaced by a capital-intensive technology when labor has become expensive. Production can start when characteristics have been defined for the product (see 3.9). If production is to order, orders can be taken immediately; deliveries have to wait until the first batch is produced. A product can be removed (see 3.8) when it no longer contributes to company profits.

3.8 Remove a product (REMOVE)

Products to be removed are selected by set selection:



When the **REMOVE** instruction is given for a product, the product is actually removed when all production orders for it have been executed. If production is both for stock and to order, all current consumer orders are reviewed in simple sequential fashion and if possible filled from stock. Products in stock that cannot be sold in this way, including all products in industries with production for stock only, are dumped without cost or income. In industries with production to order only, all existing consumer orders are filled before the product is actually removed. If a product is removed and the machines used to produce it are scrapped at the same time, the machines are actually scrapped after the current production job; consequently, the remaining production orders for this product will never be executed.

3.9 Define product characteristics (PRODP)

The screen format for the **PRODP** instruction is:

PRODUCT	DELIVERY	REORDER LEVEL	ORDER QUANT	MAXTIME	PRICE	ADV BUDGET
P1	10	2000	1000	8	200	100 000
P2	0			60	100	
P5	0			60	100	

SET PRODUCT PARAMETERS

For each product the variables can be defined:

Delivery	Quoted delivery time in days[1]
Reorder level	Level where a new production order will be given[1,2]
Orderquant	Size of production order[1,2]
Maxtime	Maximum time for a production order
Price	Selling price[1]
Adv budget	Advertising budget[1]

1 Not applicable for materials

2 Not applicable for production to order

Selling price, advertising budget, quoted delivery time and credit extended to customers (see 3.15) determine how many customers will be attracted to the product. If production is to order only, the consumer order will be stored and executed as soon as resources for production are available. When the job is finished, it will be delivered to the consumer if the quoted delivery date (i.e. the date the order was placed plus the quoted delivery time) has been met, otherwise the product will be dumped without income or extra cost. If production is for stock, a production order for the number of units defined in ORDER QUANT is given when the stock falls below the amount defined in REORDER LEVEL. As unfilled orders are subtracted from the stock, REORDER LEVEL may be set to 0 in industries where orders are accepted.

If back-orders are accepted, they are filled when a production order is finished, but only if the quoted delivery date has been met; otherwise the backorder is annulled and this will influence the market image of the producer. MAXTIME defines the maximum time allowed for production of a batch. When used in industries with production to order only, it can prevent production of orders that take too long; in industries that produce for stock, it can reduce batch size (see 2.5). Production of materials is governed by the rules for material supply (see 3.10). The product is not sold to third parties and the amount produced is immediately added to the stock of the corresponding material.

3.10 Set parameters for materials (MAT)

Data are entered for the **MAT** instruction with the following screen format:

MATERIAL	REORDER LEVEL	ORDER QUANT	REMARKS
100	4000	2000	
120	3000	1500	
150	1000	500	UNKNOWN MATERIAL

SET PARAMETERS FOR MATERIALS

The player should provide all materials needed for production, either by internal production or by appointing external suppliers. When the stock of a material falls below REORDER LEVEL, an order will be placed with a selected supplier (see 3.11) or a production order will be given. The size of this order is equal to ORDER QUANTITY. Orders are given as long as the sum of stock and unfilled orders is below the specified reorder level.

3.11 Choose suppliers (SUPPL)

Suppliers for materials are chosen with the following table instruction:

MATERIAL	SUPPLIER	SUPPLIER	SUPPLIER	SUPPLIER	REMARKS
100	S1				
120	S2	S1			INCORRECT SUPPLIER
160	ABC				

SELECT SUPPLIERS

Suppliers are chosen from a list of suppliers provided by the game administrator. For a material produced internally, the name of the company itself must be entered. If there is more than one supplier for a material, orders are given in round-robin fashion. For each supplier the list of suppliers contains the following data:

Name	Unique name of a supplier
Credit	Number of days of credit extended by the supplier
Maximum	Maximum amount of credit extended by the supplier
Discount	Discount given by supplier for immediate payment

The choice of suppliers is determined by the materials they supply and the terms for those materials. The list of materials provided by the game administrator contains the following data:

Name	Name of supplier
Material	Quality index of material
Price	Price per unit
Mean	Mean time to deliver
Std	Standard deviation of delivery time (delivery time is normally distributed)

3.12 Define terms for employees (EMPL)

The **EMP** instruction uses the following screen format:

TERM	ADVERTISE	HIRE	FIRE	WAGE
3	40 000	160	0	2500

SET TERMS FOR EMPLOYEES

The following variables are set for all employees of the company:

Term	Term (in salary periods) after which employees may leave or may be fired.
Advertise	Advertising budget for employees
Hire	Number of employees to be hired
Fire	Number of employees to be fired
Wage	Wage for a salary period

The success of a company in hiring and retaining employees depends on its advertising budget, its relative salary and its recent employment history. A long term is also appreciated by employees, but is diminishes flexibility. The game administrator will supply data, including a minimum wage, for the first round.

3.13 Ask loans (BANK)

The **BANK** instruction is a table instruction with the following format. The bank limit provides credit on call; a loan is repaid in the stated number of quarters.

BANKLIMIT	LOAN AMOUNT	QUARTERS	%	REMARKS
15 000 000	2 000 000	5	8	BANKLIMIT NOT ACCEPTED

ASK LOAN

The application for a bank limit or a loan is accepted if it is not higher than the maximum previously defined by the game administrator. A refusal is displayed in the REMARKS column. The interest percentages per year for loans and for credit on call are also defined by the game administrator. Only the first is displayed; it is used for identification of loans in the REPAY instruction (see 3.14).

3.14 Repay loans (REPAY)

Loans may be repaid before they are due. This is done with a table instruction with the following format that lists all existing loans. A loan is repaid by entering Y in the REPAY ? column.

AMOUNT	INTEREST	QUARTERS	REPAY ?
2 000 000	2	5	

LOAN REPAYMENT

3.15 Accept credit and set credit terms (CREDIT)

The first entry in the **CREDIT** instruction determines by set selection from which suppliers credit is accepted. Accepting credit from suppliers is a means of financing materials stock. Its cost can be computed from the number of days of credit and the discount for immediate payment (see 3.11). There is an upper bound to the credit extended by each supplier to a single customer; consequently, a reason not to accept credit is that demand for credit from a particular supplier would surpass the maximum amount extended.

S1 S2 S3 S4 S5 S6 S7 S8 S9 S10
ACCEPT CREDITS FROM SUPPLIERS

Second, a table instruction is used to define in which industries credit to customers is granted. Credit to customers is a marketing instrument. It entails interest cost, risk and additional clerical costs to trace customer payments as well as deliveries.

TRAD	TECHN	SPEC	ORDER
10	0	0	0

USE AND EXTEND CREDIT

4 THE REPORT

4.1 File format

INFOMARK produces an ASCII report file named "xxxnn.REP", where xxx is the company name, and nn is the number of the quarter. For example, "ABC1.REP" contains the report for company "ABC" in the first quarter. All lines in the file have the following format:

columns	length	type	attribute
1-8	8	numeric	time
9-17	9	string	code
18-26	9	string *	descr1
27-35	9	string *	descr2
36-50	15	numeric	amount
51-65	15	numeric	price

* This string may be a converted integer.

String items are left-aligned; numeric items are right-aligned with two digits after the decimal point; they contain no other formatting characters. Both contain a trailing space. Items with a null value are represented by spaces. The file may contain the following items:

name	type	explanation
machtype	string	machine type
machnr	string	'MACH' + machine number
order	string	'NR' + order numberr
process	string	process name
product	string	product name
industry	string	industry name
supplier	string	supplier name
material	string	'MAT' + material quality index
consumer	string	'C' + consumer number
employee	numeric	employee number
price	numeric	price per unit
time	numeric	number of days,months or quarters
number	numeric	number of units
amount	numeric	a sum of money

4.2 Instruction data

All instructions are copied without cost into the data file. If an instruction is always executed immediately, actual execution is not recorded. If the actual execution of the instruction may be delayed (a machine is not scrapped before it is idle) or canceled (a new machine cannot be bought if cash is unsufficient), the corresponding actual event can be recorded too. However, this will entail costs.

name	descr1	descr2	number	price	instruction
invest	machtype		$amount_1$	-	INVEST
design	product	process	-	-	DESIGN
addsup	supplier	material	-	-	SUPPL
setprod	product	$time_2$	$amount_3$	$amount_4$	PRODP
setprice	product	$time_5$	price	$amount_6$	PRODP
smat	material	-	$amount_3$	$amount_4$	MAT
planhf		-	$number_7$	$number_8$	EMPL
setsal	$time_9$		$amount_{10}$	$amount_{11}$	EMPL
limit			$amount_{12}$		BANK
getloan	$time_{13}$		$amount_{14}$	$price_{15}$	BANK
reploan	$time_{13}$		$amount_{14}$	$price_{15}$	REPAY

1) Price of machine
2) Maximum time (in days) for production run
3) Reorder level
4) Order quantity
5) Quoted delivery time in days
6) Advertising budget
7) Number of employees to be hired
8) Number of employees to be fired
9) Number of months after which employees can be fired
10) Salary per month
11) Personnel advertising budget per quarter
12) Bank limit granted
13) Number of quarters in which loan must be repaid
14) Amount of loan
15) Interest percentage per year

4.3 Events

The events that can be reported are listed in two tables; the first table contains the main events:

name	descr1	descr2	number	price	meaning
matorder	supplier	material	number		material ordered
mat	supplier	material	number	price	material delivered
nomat	supplier	material	number	price	material not delivered
paymat	supplier		amount		material paid
alloc	product	material	number		material allocated
use	product	material	number		material used
order	product	consumer	number	price	order by consumer
prodqueu	product	order	number		production order queued
prodord	product	order	number		production order
prodrdy	product	order	number		production order finished
deliver	product	consumer	number	price	product delivered
nodeliv	product	consumer	number	price	product not delivered
salepay	product	consumer	amount		payment by consumer
salary			amount		salary paid
advert	product		amount		advertising paid
removep	product		number		remove and dump product
install	machtype	machnr	amount		machine installed
noinst	machtype		amount		machine not installed
machall	machnr	order	number		machines allocated
machfree	machnr	order	number		machines freed
scrap	machtype	machnr			machine scrapped
defect	machtype	machnr			machine defective
repair	machtype	machnr	amount		machine repaired
interest		quarters	amount		interest paid
interest		quarters	0	amount	interest received
account*			amount		accounting cost paid
assignw	product	order	number		employees assigned
freework	product	order	number		employees freed
persadv			amount		personnel advertising

* Always reported without cost

The following table contains detailed reports on personnel movements:

name	descr1	descr2	number	price	meaning
hire	company+		employee		employee hired
fire			employee		employee fired
nohire			employee		employee not hired
leave	company+		employee		employee leaves
arrive	company+		employee		employee arrives
reinst			employee		fired employee reinstated
ask	company		employee		other firm hires employee

\+ Only if employee is fired or arrives from or leaves for competing company.

4.4 Inventories

The following inventories can be made:

name	descr1	descr2	number	price	meaning
mats	material		number		stock of material
ps	product				stock of product
machw	machtype	order	number		machines in use
machnu	machtype		number		machines not in use
pers			number		employees
ops	product	order	number		active operators
cash *			amount		cash balance
cash *			0	amount	overdraft

* Always reported without cost

4.5 Reporting with DBASE III

DBASE III and similar packages can be used to summarize data from the report file. First a DBASE file must be created with the following format:

Field	Field Name	Type	Width	Dec
1	TIME	Numeric	8	2
2	CODE	Character	9	
3	DESCR1	Character	9	
4	DESCR2	Character	9	
5	AMOUNT	Numeric	15	2
6	PRICE	Numeric	15	2

Next, data should be entered from the ASCII file. If the ASCII file is called ABC1.REP and the DBASE file ABC1, this is done with:

```
. use abc1
. append from abc1.rep sdf
```

Now we can either use direct statements or programs to extract data from the DBASE file.

Example 1

The following program computes a revenue and expense statement for the first or second quarter:

```
parameters filename
set talk off
close databases
use &filename
sum amount to sumexp for
code='advert'.or.code='persadv'.or.code='account'.or.code='interest'
sum amount to prodexp for
code='salary'.or.code='repair'.or.code='paymat'
sum amount to invest for code='install'
sum amount to sales for code='salepay'
sum price to suminc for code='interest'
sum amount to kas for code='cash'.and.(time=60.or.time=120)
clear
@1,0  say 'FIXED ASSETS BOUGHT  ' get invest pict '99,999,999.99'
@2,0  say 'PRODUCTION EXPENSES  ' get prodexp pict '99,999,999.99'
@3,0  say 'OTHER EXPENSES       ' get sumexp pict '99,999,999.99'
?         '                     ','-------------'
totexp=invest+prodexp+sumexp
@5,0  say 'TOTAL EXPENSES       ' get totexp pict '99,999,999.99'
@7,0  say 'SALES                ' get sales pict '99,999,999.99'
@8,0  say 'OTHER INCOME         ' get suminc pict '99,999,999.99'
?         '                     ','-------------'
totinc=sales+suminc
@10,0 say 'TOTAL INCOME         ' get totinc pict '99,999,999.99'
netinc=totinc-totexp
@12,0 say 'NET INCOME           ' get netinc pict '99,999,999.99'
@13,0 say 'CASH BALANCE         ' get kas pict '99,999,999.99'
set talk on
```

This program can be called as follows:

```
. do income with 'abc1'

FIXED ASSETS BOUGHT     4,000,000.00
PRODUCTION EXPENSES     1,562,446.85
OTHER EXPENSES            411,300.00
                        ------------
TOTAL EXPENSES          5,973,746.85

SALES                   4,241,700.00
OTHER INCOME               72,740.16
                        ------------
TOTAL INCOME            4,314,440.16

NET INCOME             -1,659,306.69
CASH BALANCE            8,340,693.31
```

Example 2

The stock of finished products at the end of the first quarter can be computed with a direct query:

```
. use abc1
. list descr1,amount for code='ps'.and.time=60
Record#  descr1              amount
    633  TRADA             7222.00
    634  TRADD              375.00
    635  SPECE             1988.00
```

However, to compute total production and sales for each product we need a more complicated program:

```
parameters filename
set talk off
set safety off
close databases
use &filename
index on descr1 to &filename
total to sales on descr1 fields amount for code='deliver'
total to prod on descr1 fields amount for code='prodrdy'
select 2
use prod
index on descr1 to prod
select 1
use sales
set relation to descr1 into prod
list descr1,prod->amount,amount
close databases
erase prod.dbf
erase sales.dbf
erase prodrdy.dbf
erase prodord.dbf
set talk on
set safety on
```

It is called with:

```
. do prodsale with 'abc1'
```

Record#	descr1	prod->amount	amount
1	SPECE	3000.00	1012.00
2	TRADA	20000.00	12778.00
3	TRADD	3000.00	2625.00

Example 3

This program shows production jobs for each product. The result is in the file batch2. Other intermediate files are deleted.

```
parameters filename
set talk off
set safety off
close databases
use &filename
copy to prodrdy for code='prodrdy'
copy structure extended to s
use s
goto 1
insert blank
replace field_name with 'STIME',field_type with 'N',field_len with  8,
 field_dec with 2
create prodord from s
use prodord
append from &filename for code='prodord'
replace all stime with time
select 2
use prodrdy
index on descr2 to prodrdy
select 1
use prodord
join with prodrdy to batches for descr2=prodrdy->descr2     fields
stime,prodrdy->time,descr1,descr2,amount
use batches
sort on descr1,stime to batch2
use batch2
list
close databases
erase s.dbf
erase prodord.dbf
erase prodrdy.dbf
erase batches.dbf
erase batch2.dbf
set talk on
set safety on
```

This program is called with:

```
. do prodrunm with 'abc1'
```

Record#	STIME	TIME	DESCR1	DESCR2	AMOUNT
1	12.64	19.64	SPECE	NR7	500.00
2	19.64	25.64	SPECE	NR8	500.00
3	25.64	31.64	SPECE	NR9	500.00
4	31.64	37.64	SPECE	NR10	500.00
5	37.64	43.64	SPECE	NR11	500.00
6	43.64	49.64	SPECE	NR12	500.00
7	21.16	29.16	TRADA	NR1	4000.00
8	29.16	35.16	TRADA	NR2	4000.00
9	35.16	43.16	TRADA	NR13	4000.00
10	43.16	51.16	TRADA	NR15	4000.00
11	51.16	57.16	TRADA	NR17	4000.00
12	7.22	14.22	TRADD	NR3	500.00
13	14.22	20.22	TRADD	NR4	500.00
14	20.22	26.22	TRADD	NR5	500.00
15	26.22	32.22	TRADD	NR6	500.00
16	33.97	40.97	TRADD	NR14	500.00
17	45.89	52.89	TRADD	NR16	500.00

5 DATA

5.1 General data

Labour market data	
Fraction of labor force departing each month	0.03
Cost of advertising to reach one worker	500
Available workers at the start of the game	500
Productivity of a new worker	0.5
Learning factor	0.3
Salary period in days	20
Minimum wage per salary period	2000

Miscellaneous data	
Starting capital in millions	10
Working days in quarter	60
Days to interest payment	20
Days for advertising budget	20
Interest percentage for credit balance *	4
Interest percentage for debit balance *	8
Interest percentage for bank loans *	8

* Interest percentages are per year (= 4 quarters)

5.2 Data for consumer industries

Industry	TRAD	TECHN	SPEC	ORDER
Minimum price	100	100	100	100
Minimum delivery time	0	10	10	15
Maximum delivery time	0	20	20	30
Stock possible	Y	Y	Y	N
Number of consumers	50	50	50	50
Reference group	low	medium	high	medium
Advertising budget	high	low	medium	medium
Average paytime	0	5	8	10
Maximum paytime	40	50	55	50

Customer data

Interarrival time				
Industry	TRAD	TECHN	SPEC	ORDER
Starting value	12	50	36	15
Reduction per period	0.01	0.015	0.02	0
Cycle	0.1	0.3	0.25	0.1
Cycle length	20	20	20	20

Average order size				
Industry	TRAD	TECHN	SPEC	ORDER
Starting value	240	1000	720	360
Growth	0	0.005	0.002	0.02
Cycle	0	0.1	0.05	0.25
Cycle length	20	20	20	20

5.3 Supplier data

Supplier	Material	Max credit		Discount	Price	Delivery time	
		Days	Amount			Mean	Stddev.
S1	100	20	1 500 000	0.01	40	10	2
S2	100	30	2 000 000	0	38	15	5
S3	100	0	2 000 000	0	42	0	0
S4	120	0	2 000 000	0	64	0	0
S5	120	20	1 500 000	0.01	60	12	2
S6	120	30	2 000 000	0	56	18	8
S7	140	20	1 500 000	0.01	80	14	3
S8	140	30	2 000 000	0	76	20	0
S9	140	0	1 500 000	0	85	0	0
S10	160	0	2 000 000	0	106	0	0
S11	160	20	1 500 000	0.01	100	15	2
S12	160	30	2 000 000	0	95	22	10

5.4 Machine data

Name	MTTD	Repair cost/day	Investment	MTBF	MTTR
M1	15	2000	2000000	60	5
M2	20	3000	5000000	200	15
M3	5	1000	500000	30	3
M4	10	800	400000	100	5
M5	22	2200	2500000	200	10
M6	7	1200	800000	25	3
M7	12	1000	600000	120	5
M8	25	3500	6000000	250	12
M9	14	900	500000	120	6
M10	23	2700	3000000	220	10
M11	12	1000	400000	110	5
M12	15	800	700000	200	10

5.5 Technology data

Type	Industry	Qual	Work	Capacity	Set-up	Materials		Machines	
						Type	Amount	Type	Nr
MAT1	MAT	160	50	20000	1	100	20000	M1	1
MAT2	MAT	140	40	50000	4	100	50000	M2	1
TRADA	TRAD	120	60	20000	1	100	15000	M3	2
						140	5000	M4	2
TRADB	TRAD	120	50	40000	2	100	15000	M3	3
						140	25000	M5	1
TRADC	TRAD	180	40	35000	3	120	15000	M5	1
						160	20000	M6	2
TRADD	TRAD	180	15	5000	1	160	5000	M4	1
TECHA	TECHN	125	70	30000	1	100	20000	M3	3
						140	10000	M7	4
TECHB	TECHN	125	50	70000	3	100	30000	M3	4
						140	40000	M8	1
TECHC	TECHN	200	40	60000	3	120	20000	M8	1
						160	40000	M6	4
TECHD	TECHN	200	12	5000	1	160	5000	M7	1
SPECA	SPEC	120	50	20000	1	100	15000	M3	2
						140	5000	M9	2
SPECB	SPEC	120	40	50000	4	100	20000	M3	3
						140	30000	10	1
SPECC	SPEC	180	40	30000	3	120	10000	M10	1
						160	20000	M6	2
SPECD	SPEC	220	20	10000	1	140	5000	M9	1
						160	5000		
SPECE	SPEC	280	15	5000	1	160	5000	M9	1
ORDA	ORDER	130	60	30000	0.2	100	20000	M3	2
						140	10000	M11	3
ORDD	ORDER	260	12	5000	0.1	160	5000	M12	1

5.6 Cost of measuring data

Event data	Inventory data	Free data
100	2000	0

6 OPERATOR'S INSTUCTIONS

6.1 Introduction

Infogame is a Turbo-Pascal (Rev 4.0) program for the IBM-Pc and compatibles. Depending on the hardware available, it can be played in four modes; multiple diskette mode and network mode are described elsewhere.

Hardware	Mode
Stand-alone PC with two diskette stations	Pilot mode
Stand-alone PC with hard disk	Standard mode
Multiple stand-alone PC's (one with hard disk)	Multiple diskette mode
PC's with disk server (Novell or similar)	Network mode

Pilot mode

Pilot mode is for experimenting on a scall scale, for example with one or two companies for one to six quarters. In this mode, program files are on A: and data files on B:. Accordingly, B: is made the default disk, and programs are called with A:<program name>. Owners of PC's without hard disk should know that mislocating files or specifying the wrong default disk is a prime cause of errors with such systems. In the present version of Infogame, the player should see that enough disk space is available, for example by copying output files from previous periods to a separate diskette. This also applies to standard mode, but it is more important with 360KB diskettes.

Standard mode

Standard mode is used with a single PC for input. If a hard disk is available, standard mode is preferable to pilot mode. Data entry is done from forms by the game administrator, or players are allocated time slots for input. All program and data files are in a dedicated partition of the hard disk. As in the pilot mode, the player should see that enough disk space is available.

6.2 Files

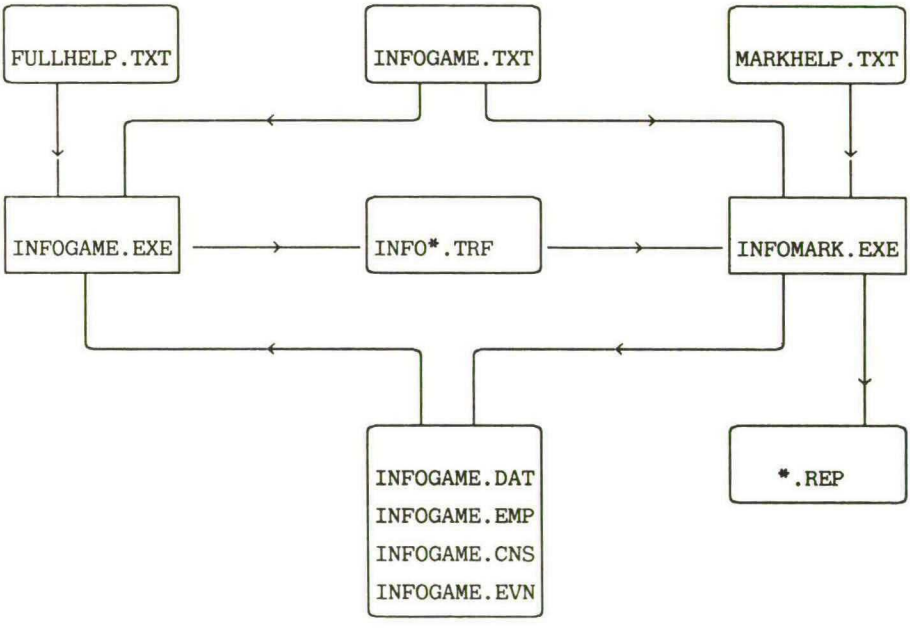
There are four types of files:

Code	Filetype	Example
P	Program	INFOMARK.EXE
D	Externally prepared data files in ASCII format	INFOGAME.TXT
I	Internal (binary coded) files	INFOGAME.DAT
T	Input and output files in ASCII format	ABC1.REP

The following table lists all files. The disk is specified for pilot mode only; in standard mode all files are in the same partition of the hard disk. The distribution disk contains the files of type P and D; the user should transfer these files to the dedicated partition of the hard disk or working diskettes before play can start.

Name	Type	Disk	Function
INFOGAME.EXE	P	A	Input program for players
INFOMARK.EXE	P	A	Program that computes the results of a round.
INFOGAME.TXT	D	B	Data supplied by game administrator
MARKHELP.TXT	D	B	Helpfile for INFOMARK
FULLHELP.TXT	D	B	Helpfile for INFOGAME
INFOGAME.DAT	I	B	General data file
INFOGAME.CNS	I	B	File with consumer data
INFOGAME.EVN	I	B	File with event data
INFOGAME.EMP	I	B	File with employee data
INFOnna.TRF	T	B	File with input data prepared by INFOGAME.
			nn = period number, a= letter (A..L)
xxxnn.REP	T	B	Report file produced by INFOMARK
			xxx= company name, nn = period number

The relation between files and programs is as follows:



6.3 Playing sequence

6.3.1 Normal play

Normal play proceeds as follows:

a) INFOMARK is called (with INFOMARK or A:INFOMARK), and the command START is executed (see 6.3.3).

b) INFOGAME is executed repeatedly (with INFOGAME or A:INFOGAME) to enter data for each player.

c) INFOMARK is called and the command NEXT is executed to compute the results of the round.

d) Players analyze data from their private *.REP files (copies of the *.REP files may be distributed to players) and proceed with b) for the next round. Even if no input changes are made, INFOGAME should be called at least once in each round.

6.3.2 Restart during play

At any moment after the first call of INFOMARK with NEXT, the game administrator can restart a previous period by a call to INFOMARK and execution of RESTART (see 6.3.3). After restart, the input for the indicated period can be corrected with INFOGAME, and the results can be computed again with INFOMARK. All input for periods after this period is lost. If all inputs for a previous period should be entered anew, the game administrator proceeds as follows: when the data for period i should be entered again, RESTART is executed for period i-1, immediately followed by NEXT. Then all input data for period i are cleared and can be entered afresh. RESTART can be used for corrections and for teaching purposes.

6.3.3 Execution of INFOMARK

Infomark uses the selection and table-filling instructions described in section 3. Like Infogame, Infomark asks for entry and/or verification of a password. After the start, an instruction is selected. When this has been executed, a new instruction can be selected; normally however, INFOMARK will be terminated by pressing F9 or ESC. The select instruction shows the list:

```
START RESTART BANKER NEXT


CHOOSE INSTRUCTION
```

The function of these instructions is:

START

A new playing session is started. After START, play starts in round 1 without any companies. If the file INFOGAME.DAT does not exist, START is the only option available. In that case the operator starts with entry and verification of a new password. If the file INFOGAME.DAT exists, the password of the current playing session has to be entered before the new session can start. This is done to prevent accidental use of START; of course, if the old password is not known, the file INFOGAME.DAT may be erased.

RESTART

Restart a preceding round in the same playing session. The appropriate session is selected from a set of options:

```
1 2 3


CHOOSE QUARTER
```

BANKER

First, the banker can select an option from a list of instructions:

MAXIMA RATES
SELECT RULE

MAXIMA

With this instruction the maximum short-time limit and the maximum amount of long term loans that can be extended are defined for all existing companies. Consequently, it can only be executed after the first round. The format of the table is:

FIRM	MAX LIMIT	MAX LOAN
ABC	5 000 000	10 000 000

SET BANK MAXIMA

RATES

With this instruction the following rates are set:
SHORT % Interest percentage (per year) for bank credit
LONG % Interest percentage (per year) for long-term loans
CREDIT% Interest percentage (per year) for deposits
The screen format for this instruction is:

SHORT %	LONG %	CREDIT %

SET BANK RATES

NEXT

Results for the next round are computed.

6.4 Changing the game administrator's data

Data in the file INFOGAME.TXT can be changed by the game administrator. The file is divided into six sections, separated by blank lines; according to Pascal conventions, strings have a fixed length of 8 characters, and numbers are separated by spaces. Changes in INFOGAME.TXT should be checked carefully; errors in this file may cause program errors.

6.4.1 General data

These cannot be changed without a thorough knowledge of the mechanics of the game; a description will be published separately.

6.4.2 Suppliers

This section contains one line for each supplier (the maximum number of suppliers is 12). Lines contain the following fields:

Supplier name
Maximum number of days of credit
Maximum amount of credit
Discount for immediate payment

6.4.3 Materials

This section contains an entry for each combination of material and supplier. The maximum number of materials is 12, so the maximum number of entries in this table is 144. Each line contains the following fields:

Name of supplier
Quality index of material
Price of material per unit
Mean delivery time
Standard deviation of delivery time (normally distributed)

6.4.4 Machines

This section contain a line for each machine type (the maximum number of machine types is 12). Each line contains the following fields:

Machine name
Mean time to deliver
Repair cost per day
Investment
Mean time between failures
Mean time to repair

6.4.5 Technologies

In this section, each technology is described by a header line, one or more lines for materials and one or more lines for machines (the maximum number of technologiews is 100). The header line contains the following fields:

Name of technology (process name).
Industry name
Quality index
Number of operators needed for production at full capacity
Production at full capacity per quarter
Startup time
Number of types of material used
Number of machine types used

Each line for a material contains the fields:
Quality index
Amount used for production at full capacity

A line for a machine type contains the fields:
Machine type
Number used for production at full capacity

6.4.6 Report data

This section contains a line for each event or state that can be reported. In the standard data, costs of all event reports and of all state reports are equal. Each line contains the following items:

Name of item
Type of report (should not be changed)
Cost of report

IN 1987 REEDS VERSCHENEN

242 Gerard van den Berg
Nonstationarity in job search theory

243 Annie Cuyt, Brigitte Verdonk
Block-tridiagonal linear systems and branched continued fractions

244 J.C. de Vos, W. Vervaat
Local Times of Bernoulli Walk

245 Arie Kapteyn, Peter Kooreman, Rob Willemse
Some methodological issues in the implementation of subjective poverty definitions

246 J.P.C. Kleijnen, J. Kriens, M.C.H.M. Lafleur, J.H.F. Pardoel
Sampling for Quality Inspection and Correction: AOQL Performance Criteria

247 D.B.J. Schouten
Algemene theorie van de internationale conjuncturele en strukturele afhankelijkheden

248 F.C. Bussemaker, W.H. Haemers, J.J. Seidel, E. Spence
On (v,k,λ) graphs and designs with trivial automorphism group

249 Peter M. Kort
The Influence of a Stochastic Environment on the Firm's Optimal Dynamic Investment Policy

250 R.H.J.M. Gradus
Preliminary version
The reaction of the firm on governmental policy: a game-theoretical approach

251 J.G. de Gooijer, R.M.J. Heuts
Higher order moments of bilinear time series processes with symmetrically distributed errors

252 P.H. Stevers, P.A.M. Versteijne
Evaluatie van marketing-activiteiten

253 H.P.A. Mulders, A.J. van Reeken
DATAAL - een hulpmiddel voor onderhoud van gegevensverzamelingen

254 P. Kooreman, A. Kapteyn
On the identifiability of household production functions with joint products: A comment

255 B. van Riel
Was er een profit-squeeze in de Nederlandse industrie?

256 R.P. Gilles
Economies with coalitional structures and core-like equilibrium concepts

257 P.H.M. Ruys, G. van der Laan
Computation of an industrial equilibrium

258 W.H. Haemers, A.E. Brouwer
Association schemes

259 G.J.M. van den Boom
Some modifications and applications of Rubinstein's perfect equilibrium model of bargaining

260 A.W.A. Boot, A.V. Thakor, G.F. Udell
Competition, Risk Neutrality and Loan Commitments

261 A.W.A. Boot, A.V. Thakor, G.F. Udell
Collateral and Borrower Risk

262 A. Kapteyn, I. Woittiez
Preference Interdependence and Habit Formation in Family Labor Supply

263 B. Bettonvil
A formal description of discrete event dynamic systems including perturbation analysis

264 Sylvester C.W. Eijffinger
A monthly model for the monetary policy in the Netherlands

265 F. van der Ploeg, A.J. de Zeeuw
Conflict over arms accumulation in market and command economies

266 F. van der Ploeg, A.J. de Zeeuw
Perfect equilibrium in a model of competitive arms accumulation

267 Aart de Zeeuw
Inflation and reputation: comment

268 A.J. de Zeeuw, F. van der Ploeg
Difference games and policy evaluation: a conceptual framework

269 Frederick van der Ploeg
Rationing in open economy and dynamic macroeconomics: a survey

270 G. van der Laan and A.J.J. Talman
Computing economic equilibria by variable dimension algorithms: state of the art

271 C.A.J.M. Dirven and A.J.J. Talman
A simplicial algorithm for finding equilibria in economies with linear production technologies

272 Th.E. Nijman and F.C. Palm
Consistent estimation of regression models with incompletely observed exogenous variables

273 Th.E. Nijman and F.C. Palm
Predictive accuracy gain from disaggregate sampling in arima - models

274 Raymond H.J.M. Gradus
The net present value of governmental policy: a possible way to find the Stackelberg solutions

275 Jack P.C. Kleijnen
A DSS for production planning: a case study including simulation and optimization

276 A.M.H. Gerards
A short proof of Tutte's characterization of totally unimodular matrices

277 Th. van de Klundert and F. van der Ploeg
Wage rigidity and capital mobility in an optimizing model of a small open economy

278 Peter M. Kort
The net present value in dynamic models of the firm

279 Th. van de Klundert
A Macroeconomic Two-Country Model with Price-Discriminating Monopolists

280 Arnoud Boot and Anjan V. Thakor
Dynamic equilibrium in a competitive credit market: intertemporal contracting as insurance against rationing

281 Arnoud Boot and Anjan V. Thakor
Appendix: "Dynamic equilibrium in a competitive credit market: intertemporal contracting as insurance against rationing

282 Arnoud Boot, Anjan V. Thakor and Gregory F. Udell
Credible commitments, contract enforcement problems and banks: intermediation as credibility assurance

283 Eduard Ponds
Wage bargaining and business cycles a Goodwin-Nash model

284 Prof.Dr. hab. Stefan Mynarski
The mechanism of restoring equilibrium and stability in polish market

285 P. Meulendijks
An exercise in welfare economics (II)

286 S. Jørgensen, P.M. Kort, G.J.C.Th. van Schijndel
Optimal investment, financing and dividends: a Stackelberg differential game

287 E. Nijssen, W. Reijnders
Privatisering en commercialisering; een oriëntatie ten aanzien van verzelfstandiging

288 C.B. Mulder
Inefficiency of automatically linking unemployment benefits to private sector wage rates

289 M.H.C. Paardekooper
A Quadratically convergent parallel Jacobi process for almost diagonal matrices with distinct eigenvalues

290 Pieter H.M. Ruys
Industries with private and public enterprises

291 J.J.A. Moors & J.C. van Houwelingen
Estimation of linear models with inequality restrictions

292 Arthur van Soest, Peter Kooreman
Vakantiebestemming en -bestedingen

293 Rob Alessie, Raymond Gradus, Bertrand Melenberg
The problem of not observing small expenditures in a consumer expenditure survey

294 F. Boekema, L. Oerlemans, A.J. Hendriks
Kansrijkheid en economische potentie: Top-down en bottom-up analyses

295 Rob Alessie, Bertrand Melenberg, Guglielmo Weber
Consumption, Leisure and Earnings-Related Liquidity Constraints: A Note

296 Arthur van Soest, Peter Kooreman
Estimation of the indirect translog demand system with binding non-negativity constraints

IN 1988 REEDS VERSCHENEN

297 Bert Bettonvil
Factor screening by sequential bifurcation

298 Robert P. Gilles
On perfect competition in an economy with a coalitional structure

299 Willem Selen, Ruud M. Heuts
Capacitated Lot-Size Production Planning in Process Industry

300 J. Kriens, J.Th. van Lieshout
Notes on the Markowitz portfolio selection method

301 Bert Bettonvil, Jack P.C. Kleijnen
Measurement scales and resolution IV designs: a note

302 Theo Nijman, Marno Verbeek
Estimation of time dependent parameters in lineair models
using cross sections, panels or both

303 Raymond H.J.M. Gradus
A differential game between government and firms: a non-cooperative approach

304 Leo W.G. Strijbosch, Ronald J.M.M. Does
Comparison of bias-reducing methods for estimating the parameter in dilution series

305 Drs. W.J. Reijnders, Drs. W.F. Verstappen
Strategische bespiegelingen betreffende het Nederlandse kwaliteits-concept

306 J.P.C. Kleijnen, J. Kriens, H. Timmermans and H. Van den Wildenberg
Regression sampling in statistical auditing

307 Isolde Woittiez, Arie Kapteyn
A Model of Job Choice, Labour Supply and Wages

308 Jack P.C. Kleijnen
Simulation and optimization in production planning: A case study

309 Robert P. Gilles and Pieter H.M. Ruys
Relational constraints in coalition formation

310 Drs. H. Leo Theuns
Determinanten van de vraag naar vakantiereizen: een verkenning van materiële en immateriële factoren

311 Peter M. Kort
Dynamic Firm Behaviour within an Uncertain Environment

312 J.P.C. Blanc
A numerical approach to cyclic-service queueing models

313 Drs. N.J. de Beer, Drs. A.M. van Nunen, Drs. M.O. Nijkamp
Does Morkmon Matter?

314 Th. van de Klundert
Wage differentials and employment in a two-sector model with a dual labour market

315 Aart de Zeeuw, Fons Groot, Cees Withagen
On Credible Optimal Tax Rate Policies

316 Christian B. Mulder
Wage moderating effects of corporatism
Decentralized versus centralized wage setting in a union, firm, government context

317 Jörg Glombowski, Michael Krüger
A short-period Goodwin growth cycle

318 Theo Nijman, Marno Verbeek, Arthur van Soest
The optimal design of rotating panels in a simple analysis of variance model

319 Drs. S.V. Hannema, Drs. P.A.M. Versteijne
De toepassing en toekomst van public private partnership's bij de grote en middelgrote Nederlandse gemeenten

320 Th. van de Klundert
Wage Rigidity, Capital Accumulation and Unemployment in a Small Open Economy

321 M.H.C. Paardekooper
An upper and a lower bound for the distance of a manifold to a nearby point

322 Th. ten Raa, F. van der Ploeg
A statistical approach to the problem of negatives in input-output analysis

323 P. Kooreman
Household Labor Force Participation as a Cooperative Game; an Empirical Model

324 A.B.T.M. van Schaik
Persistent Unemployment and Long Run Growth

325 Dr. F.W.M. Boekema, Drs. L.A.G. Oerlemans
De lokale produktiestructuur doorgelicht.
Bedrijfstakverkenningen ten behoeve van regionaal-economisch onderzoek

326 J.P.C. Kleijnen, J. Kriens, M.C.H.M. Lafleur, J.H.F. Pardoel
Sampling for quality inspection and correction: AOQL performance criteria

327 Theo E. Nijman, Mark F.J. Steel
Exclusion restrictions in instrumental variables equations

328 B.B. van der Genugten
Estimation in linear regression under the presence of heteroskedasticity of a completely unknown form

329 Raymond H.J.M. Gradus
The employment policy of government: to create jobs or to let them create?

330 Hans Kremers, Dolf Talman
Solving the nonlinear complementarity problem with lower and upper bounds

331 Antoon van den Elzen
Interpretation and generalization of the Lemke-Howson algorithm

332 Jack P.C. Kleijnen
Analyzing simulation experiments with common random numbers, part II: Rao's approach

333 Jacek Osiewalski
Posterior and Predictive Densities for Nonlinear Regression.
A Partly Linear Model Case

334 A.H. van den Elzen, A.J.J. Talman
A procedure for finding Nash equilibria in bi-matrix games

335 Arthur van Soest
Minimum wage rates and unemployment in The Netherlands

336 Arthur van Soest, Peter Kooreman, Arie Kapteyn
Coherent specification of demand systems with corner solutions and endogenous regimes

337 Dr. F.W.M. Boekema, Drs. L.A.G. Oerlemans
De lokale produktiestruktuur doorgelicht II. Bedrijfstakverkenningen ten behoeve van regionaal-economisch onderzoek. De zeescheepsnieuwbouwindustrie

338 Gerard J. van den Berg
Search behaviour, transitions to nonparticipation and the duration of unemployment

339 W.J.H. Groenendaal and J.W.A. Vingerhoets
The new cocoa-agreement analysed

340 Drs. F.G. van den Heuvel, Drs. M.P.H. de Vor
Kwantificering van ombuigen en bezuinigen op collectieve uitgaven 1977-1990

341 Pieter J.F.G. Meulendijks
An exercise in welfare economics (III)

342 W.J. Selen and R.M. Heuts
A modified priority index for Günther's lot-sizing heuristic under capacitated single stage production

343 Linda J. Mittermaier, Willem J. Selen, Jeri B. Waggoner, Wallace R. Wood
Accounting estimates as cost inputs to logistics models

344 Remy L. de Jong, Rashid I. Al Layla, Willem J. Selen
Alternative water management scenarios for Saudi Arabia

345 W.J. Selen and R.M. Heuts
Capacitated Single Stage Production Planning with Storage Constraints and Sequence-Dependent Setup Times

346 Peter Kort
The Flexible Accelerator Mechanism in a Financial Adjustment Cost Model

347 W.J. Reijnders en W.F. Verstappen
De toenemende importantie van het verticale marketing systeem

348 P.C. van Batenburg en J. Kriens
E.O.Q.L. - A revised and improved version of A.O.Q.L.

349 Drs. W.P.C. van den Nieuwenhof
Multinationalisatie en coördinatie
De internationale strategie van Nederlandse ondernemingen nader beschouwd

350 K.A. Bubshait, W.J. Selen
Estimation of the relationship between project attributes and the implementation of engineering management tools

351 M.P. Tummers, I. Woittiez
A simultaneous wage and labour supply model with hours restrictions

352 Marco Versteijne
Measuring the effectiveness of advertising in a positioning context with multi dimensional scaling techniques

353 Dr. F. Boekema, Drs. L. Oerlemans
Innovatie en stedelijke economische ontwikkeling

354 J.M. Schumacher
Discrete events: perspectives from system theory

355 F.C. Bussemaker, W.H. Haemers, R. Mathon and H.A. Wilbrink
A (49,16,3,6) strongly regular graph does not exist

356 Drs. J.C. Caanen
Tien jaar inflatieneutrale belastingheffing door middel van vermogensaftrek en voorraadaftrek: een kwantitatieve benadering